# Căn bản về GIT

Các hệ quản lý mã nguồn (SCM) khác sẽ lưu file mã nguồn và thông tin thay đổi (delta change)



Các SCM khác Git lưu nội dung thay đổi (delta change) so với file mã nguồn ban đầu

Việc lưu delta change này cần phải sử dụng phần mềm làm server để tập trung, đánh chỉ mục các thay đổi thuộc một phiên bản.

GIT trong mỗi phiên bản sẽ lưu những file thay đổi, và liên kết trỏ về những file không thay đổi từ phiên bản trước đó



Mỗi phiên bản lưu luôn file được cập nhật và link liên kết đối với những file không thay đổi. Ví dụ ở Version 2 và 3, file B không thay đổi

Cách quản lý này, khiến GIT hoạt động giống hệ thống quản lý thư mục trong hệ Linux. Khả năng *đặt đâu cũng chạy* tốt hơn hẳn các SCM khác. Hầu như mọi hoạt động thay đổi file đều diễn ra ngay trên lưu trữ cục bộ (local repository), cho đến khi nào dev muốn thực sự chốt thay đổi (commit) lên remote repository.

Git so sánh sự thay đổi, cũng như đánh chỉ mục file như thuật toán băm SHA-1 tạo ra một chuỗi dài 40 ký tự. Mọi hành động trên GIT luôn là bổ xung thêm dữ liệu, chứ không xóa đi, do đó, khả năng quay lui phục hồi dữ liệu rất cao. Nhưng nhược điểm của GIT là kích thước Git repository sẽ tăng nhanh chóng. Giá ổ cứng ngày càng giảm, do đó bạn không phải quá lo vấn đề này.

# Ba trạng thái của file mã nguồn

1. Committed: file ở thư mục làm việc (working directory) đã đồng nhất với file được lưu trong local repository
2. Modified: file đang thay đổi
3. Staged: file thay đổi, được chuẩn bị để chốt hạ (commit) vào local repository lần tới. Nếu commit thành công, thì file lại chuyển về trạng thái Commited.

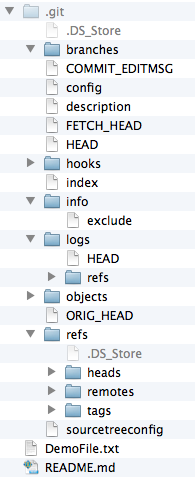




Một hình ảnh nữa mô tả rõ hơn về 3 trạng thái

Học lập trình trực tuyến cấp tốc

Thư mục .git là thư mục ẩn



git repository khi mở ra chưa nhiều thư mục con

Nếu Git chưa cài thì có 2 cách để cài.  
**Trường hợp 1**: Cài XCode sau đó cài Command Line Tools (mở XCode > Preferences… chọn tab Downloads để kiểm tra).  
Khi này Command Line Tools tự động cài Git client ở thư mục /usr/bin

**Trường hợp 2**: Bạn không dùng XCode để lập trình mà dùng tool khác để lập trình. Khi này cần cài đặt Git từ [Bộ cài Git cho MacOSX ở đây](http://git-scm.com/download/mac). Git sẽ được cài ở thư mục /usr/local/git. File chạy thực sự sẽ ở /usr/local/git/bin/git

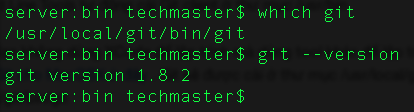
Thường thì version git do XCode command line tools thấp hơn bản Git mới nhất tải về từ http://git-scm.com/. Nếu bạn thích dùng phiên bản mới hơn thì phải sửa biến $PATH để ưu tiên tìm git ở /usr/local/git trước.  
1- Mở Terminal, gõ lệnh edit ~/.bash\_profile  
2- File .bash\_profile sẽ được mở bằng trình soạn text mặc định trong Mac của bạn. Bổ xung các dòng sau đây

GIT\_HOME=/usr/local/git

PATH=${GIT\_HOME}/bin:${PATH};export PATH

MANPATH=${GIT\_HOME}/share/man:${MANPATH};export MANPATH

3- Đóng Terminal, mở lại rồi gõ which git  
xem kết quả có phải là thư mục /usr/local/git/bin/git không?



Ví dụ kết quả trong máy của tôi

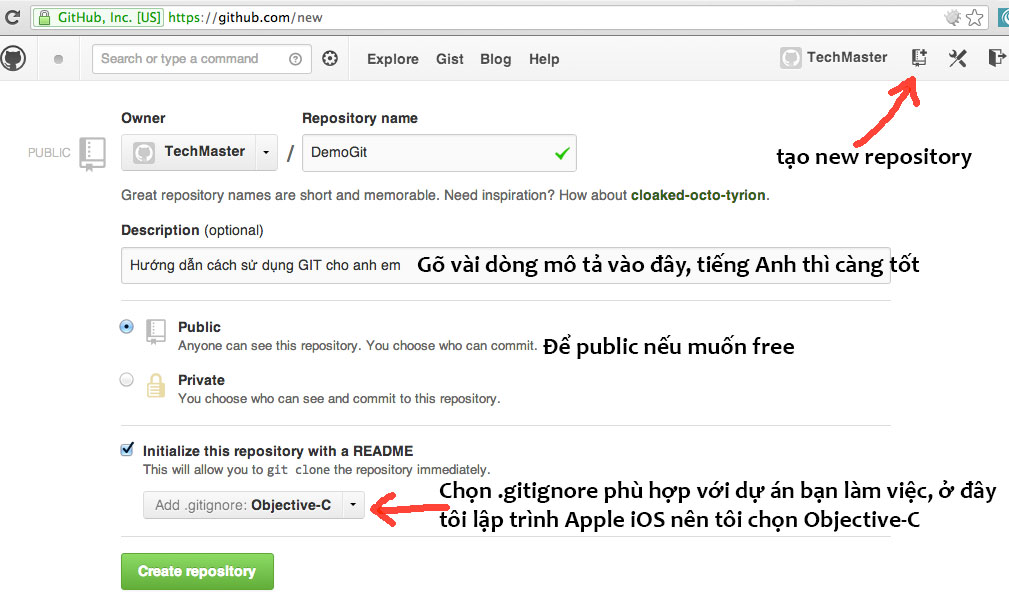
**Một số dịch vụ quản lý mã nguồn miễn phí, hoặc có một phần miễn phí.**  
Cá nhân tôi thích xài [https://bitbucket.org](https://bitbucket.org/) để lưu dự án với nhóm 5 lập trình, đặt Repository ở chế độ không công khai (private) mà không phải trả phí. Rất phù hợp cho nhóm cộng tác nhỏ, làm dự án gia công.

Ngoài ra còn một dịch vụ nổi tiếng khác là [http://github.com](http://github.com/), để miễn phí bắt buộc bạn phải công khai, chia sẻ mã nguồn.

Thực ra việc chia sẻ là có lợi, mã nguồn hay được nhiều người ngó đến, xuất sắc, bạn sẽ có nhiều cơ hội làm ăn hơn, thậm chí được một công ty Mỹ mời qua làm. Tôi đã biết vài trường hợp như vậy, CV chỉ cần viết dăm dòng link đến mã nguồn cool mà bạn để trên github. Còn hơn chán nhiều chú phải bỏ rất nhiều tiền sang My du học CS, IT rồi cày cục kiếm việc làm. Nếu bạn thực sự thông minh hay chia sẻ code hay trên GitHub !

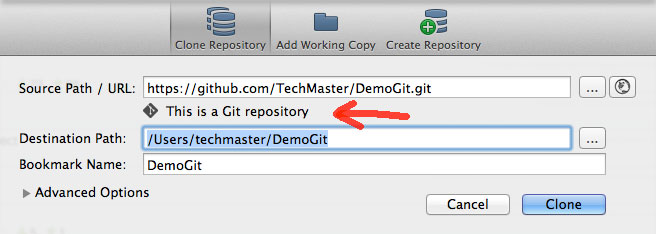
Trong những phần trước tôi đã hướng dẫn các bạn phần nào một [lập trình viên](https://techmaster.vn/khoa-hoc/25485/lap-trinh-python-ung-dung-thuc-te) thao tác với GitHub. Trong phần này tôi sẽ giới thiệu kết nối vào GitHub, tạo tài khoản và sử dụng SourceTree cho trực quan. Bài sau sẽ quay sang dùng lệnh thao tác.

Tạo mới một Repository trên GitHub

Tạo mới một Repository trên GitHub. Click vào để xem rõ hơn

[Tải về phần mềm SourceTree của Atlassian ở đây](http://www.sourcetreeapp.com/)

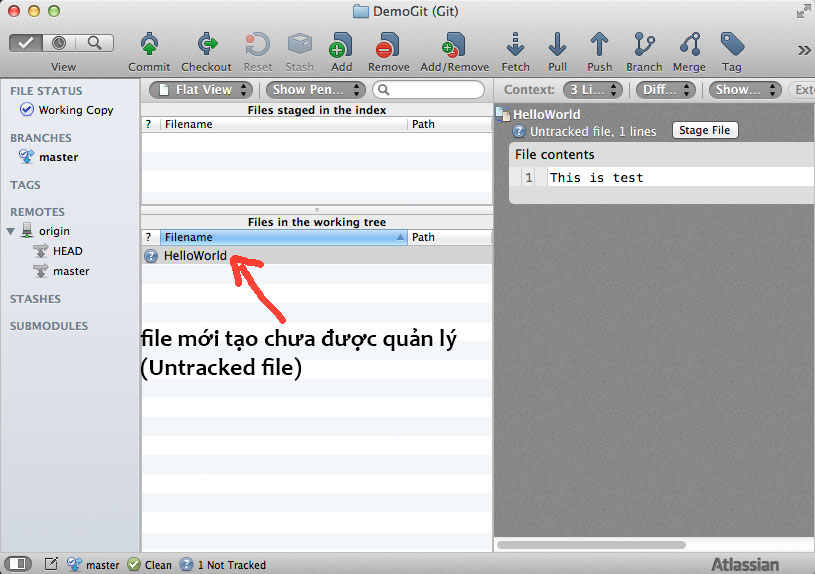
Trong SourceTree, tiến hành Clone Repository từ GitHub về thư mục local

Clone một Repo trên GitHub nhớ SourcePath/URL phải đúng thì mới được

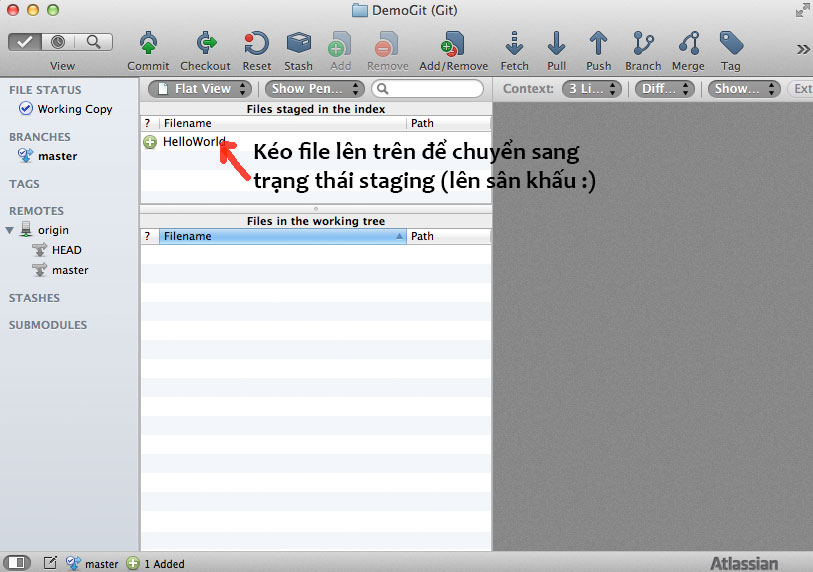
Lưu ý thư mục DemoGit phải hoàn toàn rỗng thì mới có thể Clone được về. Nếu không SourceTree sẽ báo lỗi sau “Cannot Clone Into Non-Empty Folder”. Dưới đây là thư mục DemoGit ở local khi clone về, ta có thể xem được cả file ẩn nhờ gõ lệnh  
defaults write com.apple.Finder AppleShowAllFiles YES; killall -HUP Finder  
Nếu muốn giấu file ẩn thì lại gõ lệnh  
defaults write com.apple.Finder AppleShowAllFiles NO; killall -HUP Finder  
Trong thư mục này có file .gitignore để bảo git bỏ qua một số file do ứng dụng tự sinh, không cần thiết phải quản lý.

Thư mục DemoGit sau khi clone về. Tôi đã bật lựa chọn xem cả file hiddenThư mục DemoGit sau khi clone về.  
Tôi đã bật lựa chọn xem cả file hidden

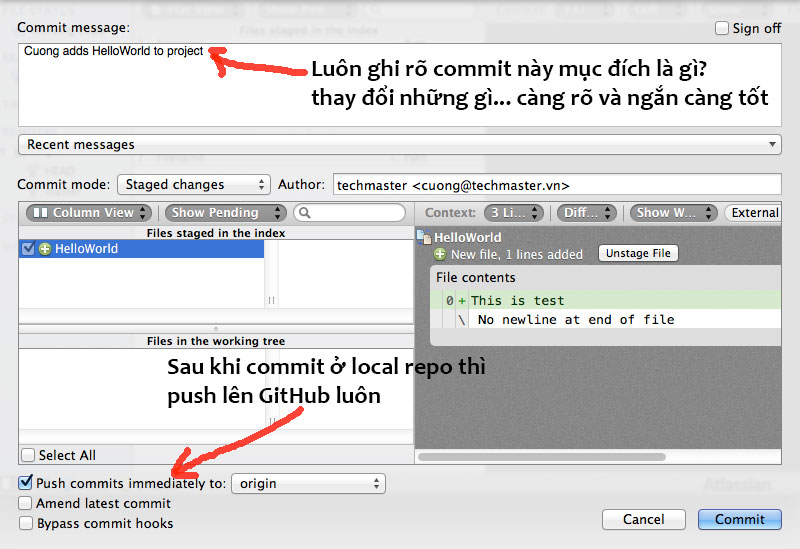
Tạo thử một file text HelloWorld trong thư mục DemoGit. Mở SourceTree ta sẽ thấy



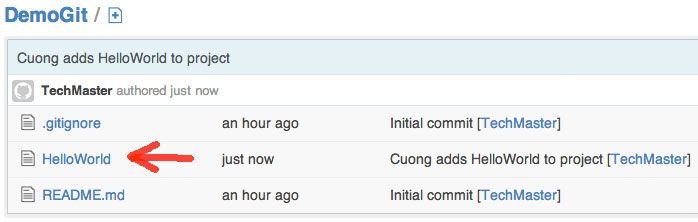
Kéo file này lên phía trên để chuẩn bị commit, file lúc này sẽ ở trạng thái staging (lên sâu khấu)



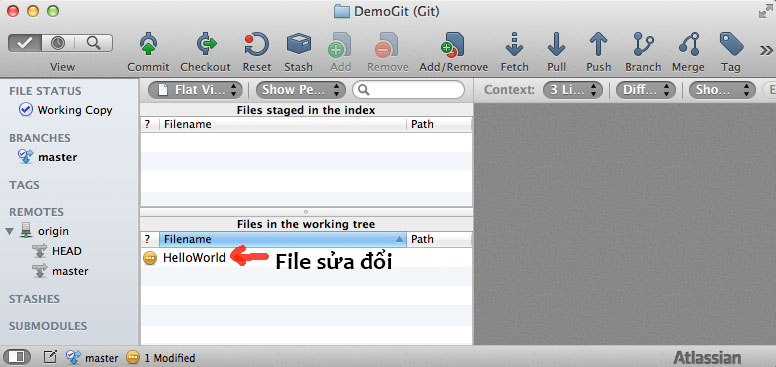
Ấn nút Commit để chốt hạ code vào Local Repo.

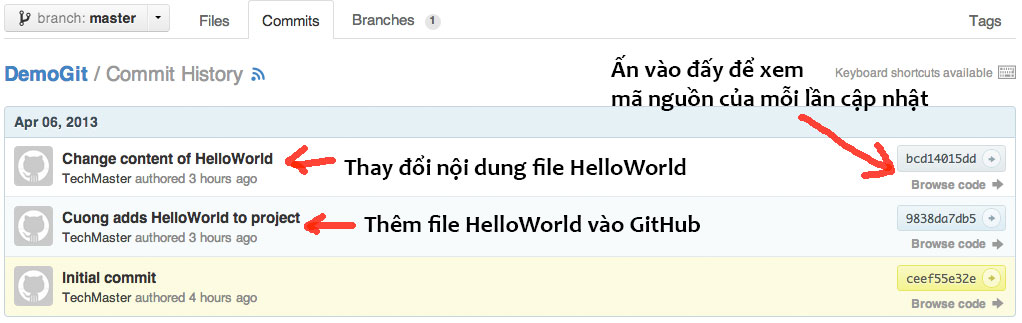
Commit ở local repo sau đó push lên GitHub server

Kết quả đây

File sau khi đã push lên GitHub có thể xem qua web

Sửa file bằng TextEdit hoặc TextWrangler, mở SourceTree sẽ thấy file HelloWorld được đánh dấu là Modified.

Sửa đổi file, trong SourceTree sẽ hiện ra file nào sửa đổi so với commit gần đây

Mở browser để xem mỗi lần sửa đổi

# Phần 2 t demo

## Create a “repository”

* Truy cập vào trang git(github, gitlab, bitbuckit)
* Click new repository
* Đặt tên cho repository

## Clone repository

* cop